

WEEKLY REPORT

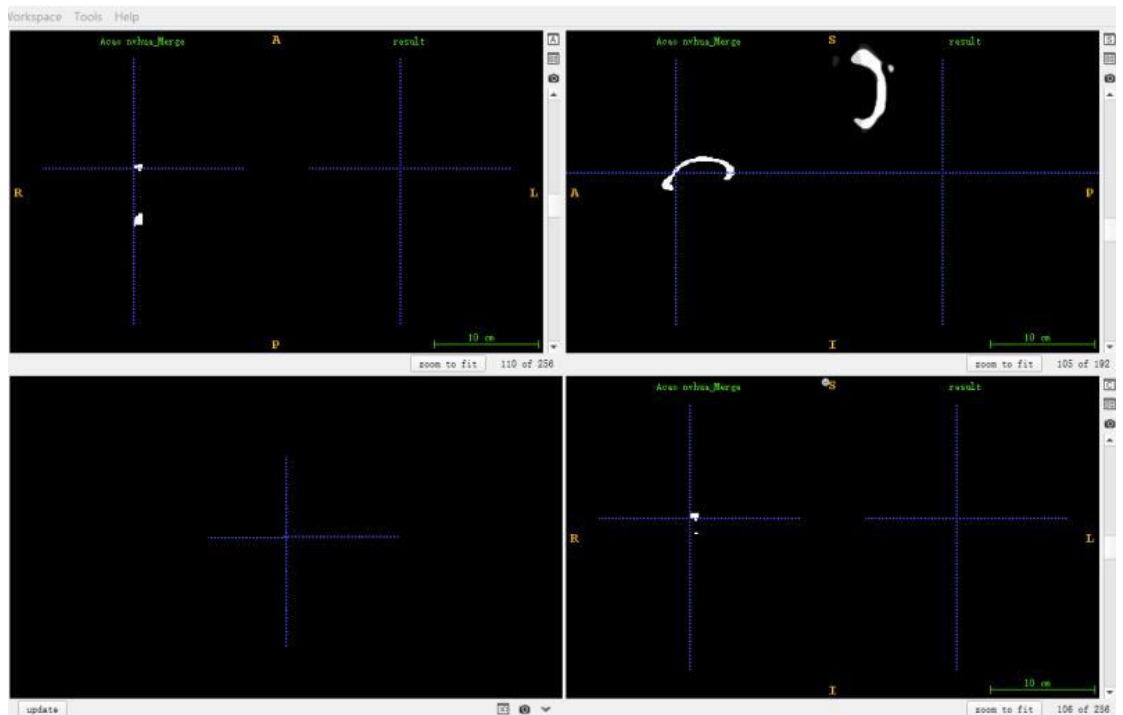
严凡

Mararch 17,2019

1. 上周工作

1.1 调整自己的 u-net 分割结果

1. 上周由于自己的训练准确率只有 0.02 并且结果一片黑，所以这周试了下上次开会提到的一些优化方案。排除特定低通道的图片。然后成功的训练出不错的结果了。



精确度达到百分之 65

1.2 跑 NF 数据

训练出结果后我就开始研究学长的代码了。希望能够从中提取些东西改善自己的代码。而且跑数据也并不能直接拿来用。也需要我看懂后对代码做一定的微调才能上手

```

File "/home/yf/anaconda3/envs/medical/lib/python3.6/site-packages/tensorflow/python/client/session.py", line 1348, in _do_call
    raise type(e)(node_def, op, message)
tensorflow.python.framework.errors_impl.InvalidArgumentError: Key: image/shape. Can't parse serialized Example.

[[{{node DecodeProto/ParseSingleExample/ParseSingleExample}} = ParseSingleExample[Tdense=[DT_INT64, DT_STRING, DT_STRING, DT_INT64, DT_STRING, DT_INT64], dense_keys=["extra/bbox", "image/encoded", "image/name", "image/shape", "segmentation/encoded", "segmentation/shape"], dense_shapes=[[6], [], [], [4], [], [3]], num_sparse=0, sparse_keys=[], sparse_types=[], _device="/device:CPU:0"]](arg0, DecodeProto/ParseSingleExample/Const, DecodeProto/ParseSingleExample/Const_1, DecodeProto/ParseSingleExample/Const_1, DecodeProto/ParseSingleExample/Const, DecodeProto/ParseSingleExample/Const_1, DecodeProto/ParseSingleExample/Const)]

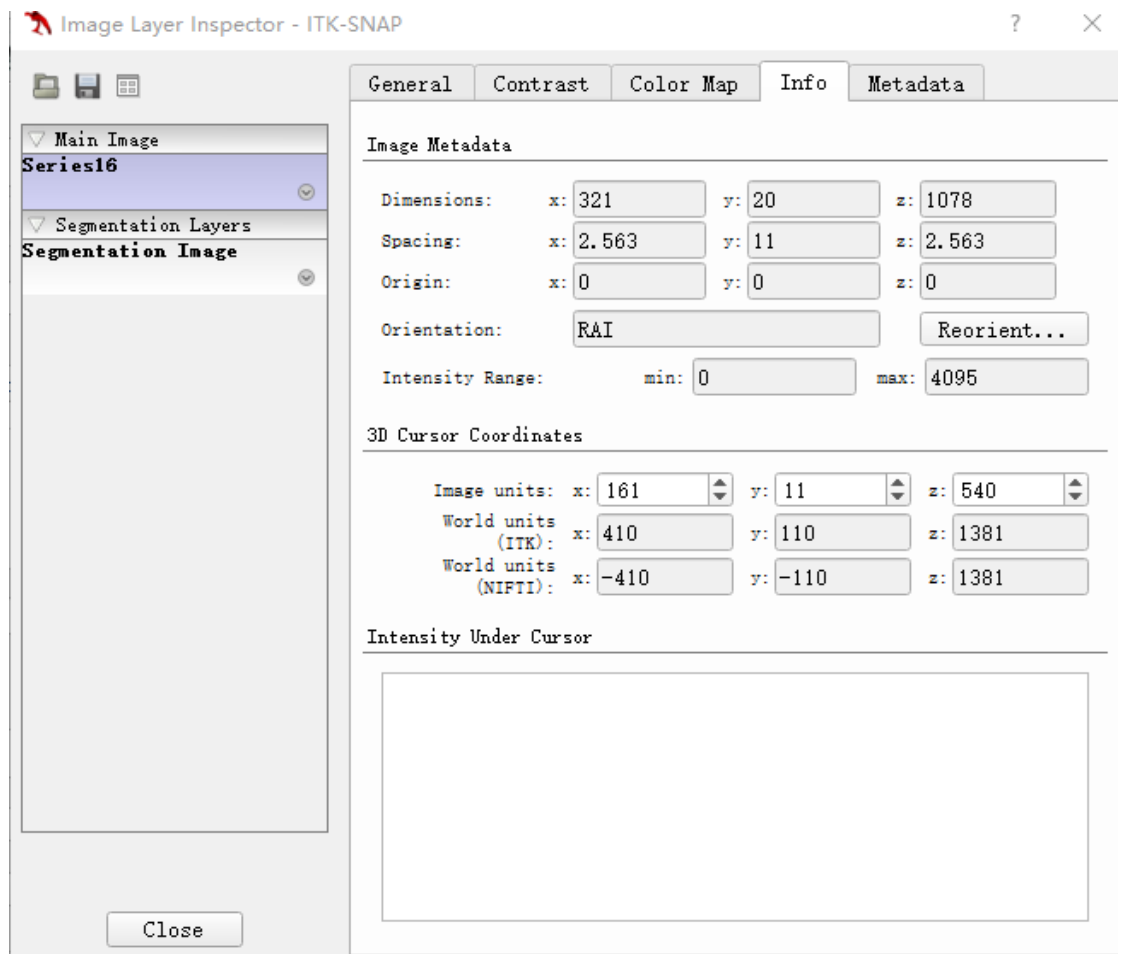
[[node IteratorGetNext (defined at /home/yf/PythonProject/MedicalImageSegmentation/custom_estimator.py:143) = IteratorGetNext[output_shapes=[[?,6], [?,128,512,?], [?], [?], [?,128,512]], output_types=[DT_INT64, DT_FLOAT, DT_STRING, DT_INT64, DT_INT32], _device="/job:localhost/replica:0/task:0/device:CPU:0"]](IteratorFromStringHandleV2)]

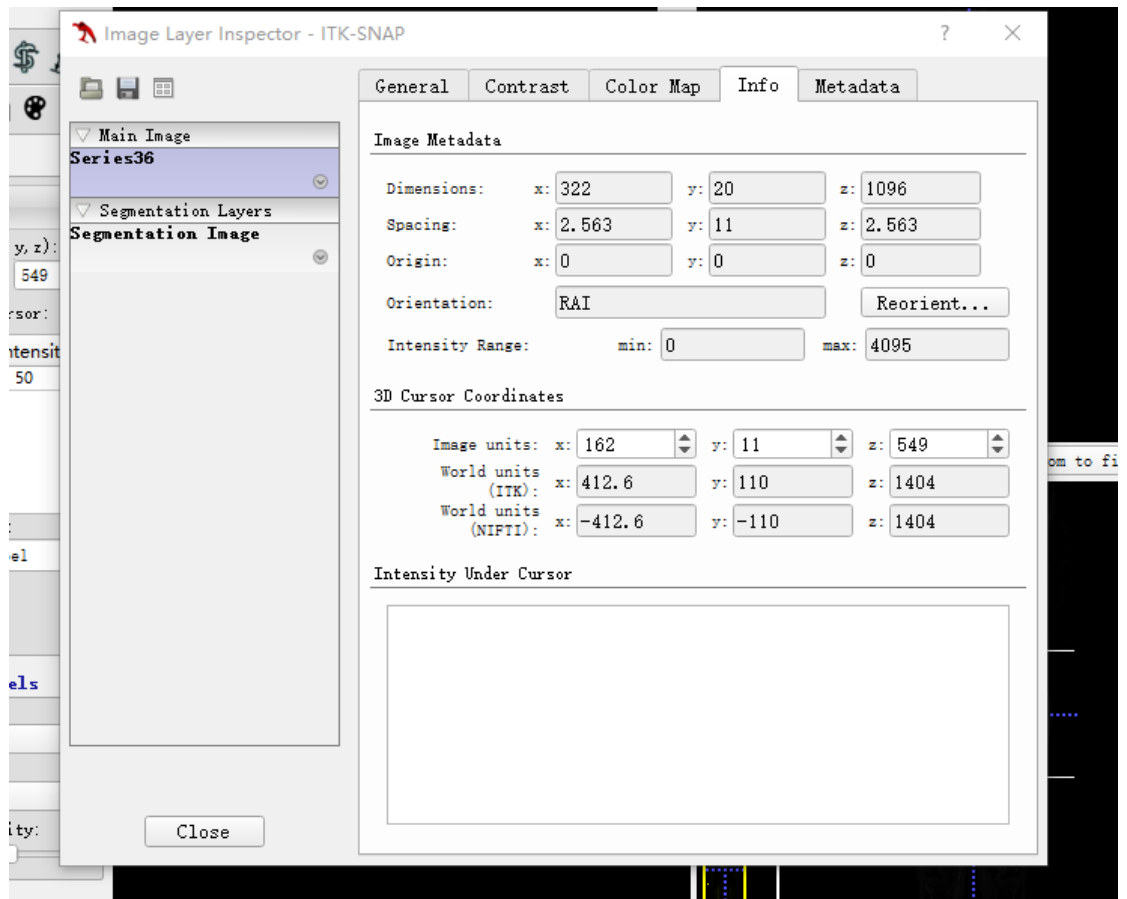
[[{{node IteratorGetNext/_1889}} = _Recv[client_terminated=false, recv_device="/job:localhost/replica:0/task:0/device:GPU:0", send_device="/job:localhost/replica:0/task:0/device:CPU:0", send_device_incarnation=1, tensor_name="edge_71_IteratorGetNext", tensor_type=DT_FLOAT, _device="/job:localhost/replica:0/task:0/device:GPU:0"]()]]
(medical) yf@mgh3dmidea25nc:~/PythonProject/MedicalImageSegmentation$

```

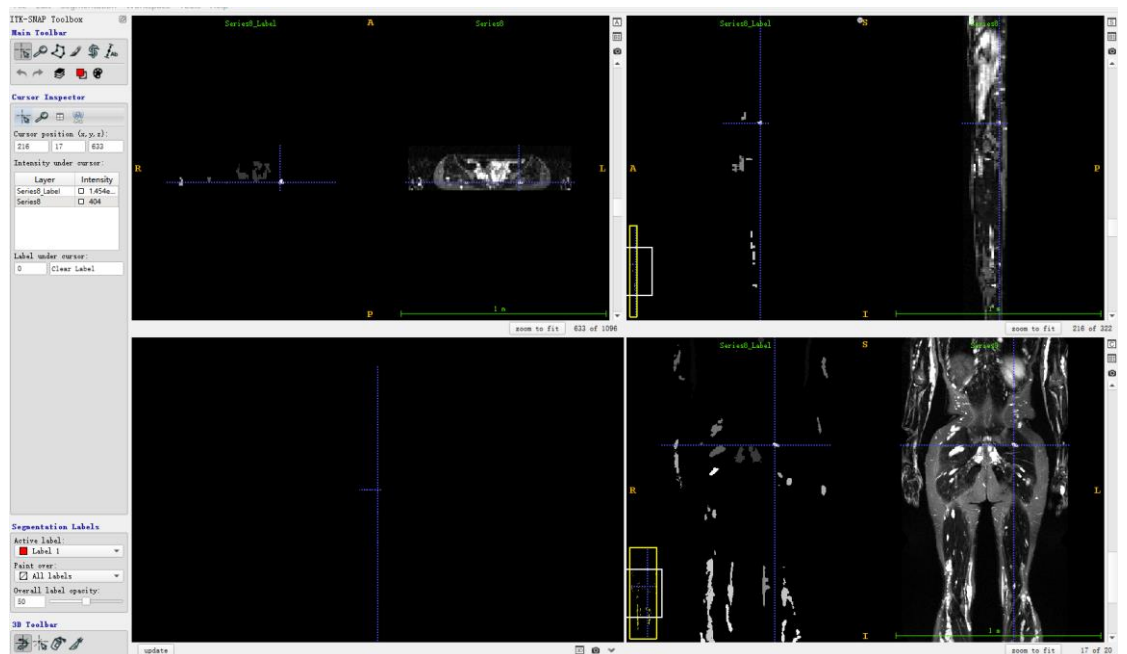
目前存在的问题有

1. 数据的宽度大小不定





2. 样本的结果标记多样，我需要把样本结果的值统一起来



3. 但是代码量有点多，来不及理解。正在尽可能的挑核心区域理解，修改

MedicalImageSegmentation [E:\PythonProject\MedicalImageSegmentation] - ...main.py [MedicalImageSegmentation] - PyCharm

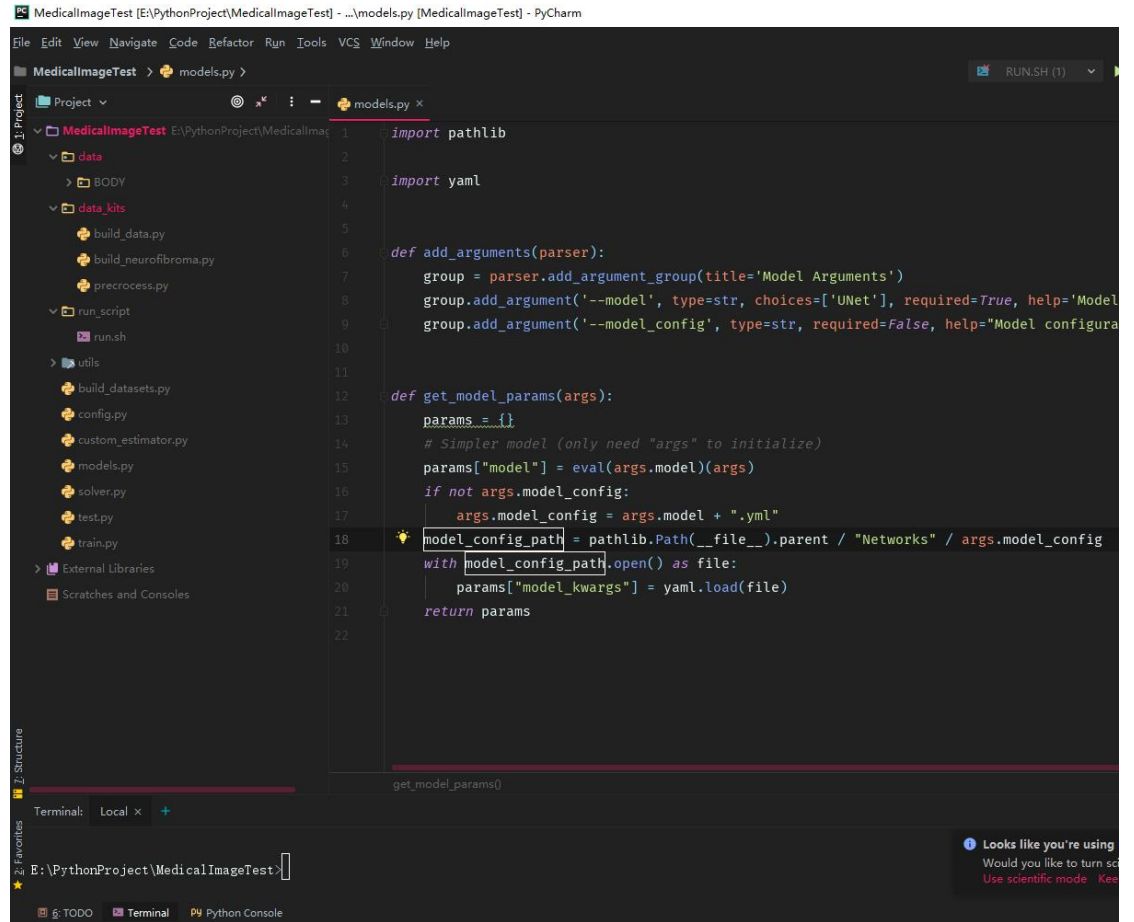
File Edit View Navigate Code Refactor Run Tools VCS Window Help

MedicalImageSegmentation > main.py > BUILD_DATASETS

Project > MedicalImageSegmentation E:\PythonProject\MedicalImageSegmentation

- data
 - BODY
 - records
 - k_folds.txt
 - trainval.txt
 - dataset_example.json
 - eval_dataset.json
 - README.md
 - train_dataset.json
 - data_kits
 - __init__.py
 - analysis_kits.py
 - analyze_lits.py
 - build_data.py
 - build_lits_liver.py
 - build_mgh_abdomen.py
 - build_neurofibroma.py
 - preprocess.py
 - Networks
 - Backbone
 - __init__.py
 - base.py
 - DeepLabV3Plus.py
 - DeepLabV3Plus.yml
 - UNet.py
 - UNet.yml
 - UNet_Small.yml
 - run_scripts
 - 004_triplet.sh
 - body.sh
 - utils
 - __init__.py
 - array_kits.py
 - array_kits_test.py
 - image2mhd.py
 - image_ops.py
 - image_ops_test.py
 - logger.py
 - mhd_kits.py
 - nii_kits.py
 - summary_kits.py
 - surface.py
 - test.py
 - timer.py
 - visualization
 - .gitignore
 - build_datasets.py
 - config.py
 - custom_estimator.py
 - custom_evaluator.py
 - custom_evaluator_base.py
 - custom_hooks.py
 - input_pipeline.py
 - input_pipeline_test.py
 - loss_metrics.py

```
75 if args.mode == ModeKeys.TRAIN:
76     log_step_count_steps = 500
77     run_config = tf.estimator.RunConfig(
78         tf_random_seed=TF_RANDOM_SEED,
79         save_summary_steps=200,
80         save_checkpoints_steps=5000,
81         session_config=_get_session_config(),
82         keep_checkpoint_max=1,
83         log_step_count_steps=log_step_count_steps,
84     )
85
86     params = {"args": args}
87     params.update(models.get_model_params(args))
88     params.update(solver.get_solver_params(args))
89     if not args.train_without_eval:
90         params.update(custom_evaluator.get_eval_params(eval_steps=args,
91                                                         primary_metric=
92                                                         secondary_metric=
93
94     estimator = CustomEstimator(models.model_fn, args.model_dir, run_c
95
96     steps, max_steps = ((args.num_of_steps, None)
97                         if args.num_of_steps > 0 else (None, args.num_
98     estimator.train(input_pipeline.input_fn,
99                     hooks=[LogLearningRateHook(tag=args.tag,
100                                                every_n_steps=log_step_
101                                                output_dir=args.model_d
102                     steps=steps,
103                     max_steps=max_steps,
104                     save_best_ckpt=args.save_best)
105
106 elif args.mode == ModeKeys.EVAL:
107     run_config = tf.estimator.RunConfig(
108         tf_random_seed=TF_RANDOM_SEED,
109         session_config=_get_session_config()
110     )
111
112     params = {"args": args}
113     params.update(models.get_model_params(args))
114
115     estimator = CustomEstimator(models.model_fn, args.model_dir, run_c
116
117     predict_keys = None
118     evaluator = EvaluateVolume(estimator, predict_keys)
119
120     estimator.evaluate(evaluator,
121                       input_pipeline.input_fn,
122                       checkpoint_path=args.ckpt_path,
123                       latest_filename=("checkpoint_best" if not args.
124                       cases=args.eval_num)
125
126
127 if __name__ == "__main__":
128     tf.app.run(main)
129
130 main() > if args.mode == ModeKeys.TRAIN
```



下周任务

1. NF 数据跑通

工作时长(10+6+4+6+8+6+8)=48 小时